Poster Abstract: Pair-wise Resistance to Traffic Analysis in MANETs

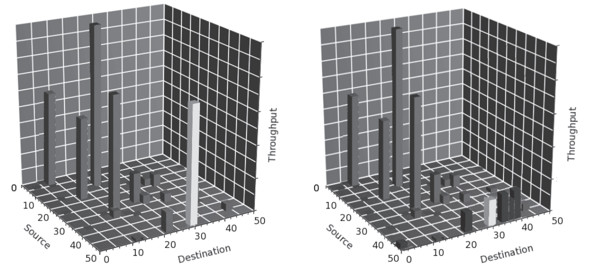
Chase Gray, Jason Byrnes, Srihari Nelakuditi
{graycm, byrnesj, srihari}@cse.sc.edu
Department of Computer Science and Engineering
University of South Carolina

I. Introduction and Motivation

Some of the same features that make MANETs attractive, such as mobility and self-organization, also lead to increased vulnerability to traffic analysis. Data on who is communicating with whom, how often, how much, and when is easily available to any eavesdropper within range of the wireless network. Even if the payload is encrypted, standard MANET protocols transmit enough header and routing information in the clear making traffic analysis relatively easy for attackers. But users of MANETs may want to resist traffic analysis for a variety of reasons, ranging from secrecy for government and industry to simple personal privacy for individuals. Traffic analysis is a threat to secure communication, either by identifying targets for attacks such as denial-of-service or encryption cracking, or by revealing communication relationships.

Network-wide Anonymity Perhaps the most enduring idea for resisting traffic analysis is the "mix" [1], in which messages are routed through an unpredictable series of proxies that alter it at each hop via encryption, stalling, padding, and other means designed to prevent recognition. Onion routing [2] is a derivative of the mix in which messages are encrypted in layers that must be decrypted independently by routers at each hop toward the destination. Another classic approach, Crowds [3], has cooperating nodes forward messages for each other in an unpredictable manor amongst themselves before sending the message toward its destination. While the above approaches are applicable to both wired and wireless, other efforts have produced protocols that are specifically designed for multi-hop wireless networks [4]. All these protocols require all or a large number of the nodes in the network to implement the protocol.

Pair-wise Anonymity For governments and corporations that control their own networks, network-wide anonymity protocols may be sufficient. However, there are other scenarios where a pair of users (government, industrial, or private) must communicate in a *public* network, but still wish to keep their relationship hidden. A public network will be unlikely to have any sort of anonymity protocol since it increases complexity and cost, and is simply not needed by most users. Therefore, we need to devise a new protocol for enabling private anonymous communication between a pair of nodes without a network-wide anonymity protocol, which is the focus of this paper.



(a) before (b) after

Figure 1: Perceived throughput for each pair of nodes in a MANET *before* and *after* source node 48 takes measures to resist traffic analysis.

To demonstrate users' vulnerability to traffic analysis, we simulated in ns2 50 randomly deployed nodes with 30 TCP and UDP flows of random duration. We selected a node pair to be the "communicators of interest" and set a 10-second UDP flow between them. Figure 1(a) shows the information an omniscient attacker would be able to obtain from silently observing the network. The communication from nodes 48→30 shows up as one of the higher-throughput flows, so an attacker can assume a relationship between the two. Though not shown in the figure, this throughput could then be shown over time to show the duration, frequency, and volume at different times.

We can resist traffic analysis by altering the apparent destinations of packets which changes an adversary's view from Figure 1(a) to that of Figure 1(b). Node 48 still appears to be sending the same volume of traffic, but the apparent destinations are more numerous and thus each flow is less significant. Now, an adversary would be more likely to focus attention on other more significant flows. Our goal is to enable 48 and 30 to accomplish this effect without the assistance or knowledge of other nodes in the network.

II. Our Approach

A pair of nodes can resist traffic analysis as illustrated in the previous section by having the source node address packets to another node in the network such that the path passes *through* or *near* the intended destination. The intended destination listens to *all* transmissions in its neighborhood and extracts those packets that are meant for it. We refer to this approach as *Co*vert *N*eighborhood *Tra*nsmissions (CoNTra). Our approach takes advantage of the *broadcast* transmissions and *multi-hop* paths intrinsic to MANETs.

Suppose node S in Figure 2 wishes to reduce or eliminate the perceived volume of traffic it is sending to D. S constructs and sends a packet that has data encrypted for D but a destination address for some other node, such as 6. If the packet's route is S→1→4→6, when 4 transmits to 6, D can overhear the transmission and receive the packet. Therefore, D must listen to *every* transmission it can hear and check if the packet is really a CoNTra packet intended for itself.

Analyzer Model Before describing how we decide where to send covert packets we need to define a model of the attacker. We assume an omniscient observer that can hear all packet transmissions but may or may not be aware of CoNTra. The best traffic analysis strategy for the CoNTra-aware observer would be to count transmissions at all nodes that can somehow receive a packet from S (directly, as a forwarder, or by overhearing) and suspect all of these nodes as CoNTra recipients of S. It could then narrow down these suspects by observing how the set of suspects changes as nodes move, join, and leave the network.

As an example, suppose S is sending CoNTra packets to D on the path S→1→4→6. A naïve analyzer would count all the traffic from S as messages for 6. A CoNTra-aware analyzer, however, would equally suspect 1, 2, 3, 4, D, and 6; that is, all the nodes that can hear a transmission anywhere on the path from S to 6. Node 6 is included as a suspect because the CoNTra-aware analyzer can't be sure if CoNTra is being used or if this is just a normal transmission. Now, instead of knowing precisely who the receiver is, there are 6 possible nodes to choose from. Now, suppose node 1 leaves the network and the CoNTra path switches to S→2→4→6. The analyzer will observe the same volume on this new path, so can therefore eliminate 1 from its list of suspects. Also note that although 5 is now in overhearing range of the path, the clever analyzer would not add it to the suspect set because it was not a potential receiver before 1 left. Therefore the number of suspects is narrowed from 6 to 5.

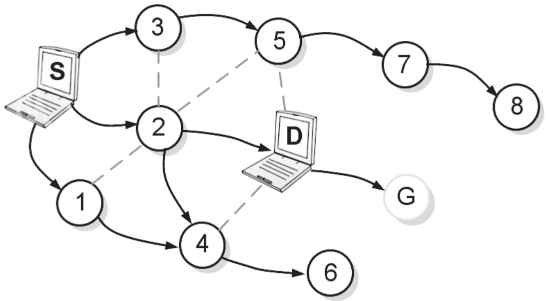


Figure 2: A sample network: S is the CoNTra sender, D is the destination, and G is a ”ghost node”. Solid lines indicate direct transmissions and dashed lines indicate overheard transmissions.

Note that realistic traffic analysis will not be as easy as the above example due to network dynamism and uncertainty about packet receptions. Also, in realistic situations attackers would only have limited view of the overall network so would have to perform their analysis with incomplete data. Finally, S will likely have non-CoNTra flows to other destinations. Since it is difficult to impossible for an observer to distinguish CoNTra from non-CoNTra traffic it would have to suspect nodes along these other paths as well.

Path Selection The first step in CoNTra path selection is to find paths that either pass through D or through a neighbor that D can overhear. S first identifies D's neighbors, then compiles a list of paths that include but do not terminate at the neighbor or D itself. The neighbor and path information can be gathered in a variety of ways, depending on the routing protocol. In DSR, for example, a node maintains a path cache for not only paths that it is using but for all overheard paths. D may assist S by sending it neighbors and potentially useful destinations derived from its cache. D can further improve the probability of successful transmission by using signal-to-noise ratio to identify neighbors that it can hear consistantly and sending only the best candidates to S.

The next step is to choose the best path from the list of paths compiled in the first step. To select this path S again uses any information it has about the network to calculate which path involves the largest number of nodes, through either overhearing or inclusion on the path itself. Under the global attacker model, the best strategy is to select one path for CoNTra that routes through as many nodes as possible and use this path as long as the topology is consistent.

It may be tempting to use multiple paths to widen the set of potential suspects. But the flaw in this approach is that no matter which path is added, D will always overhear the highest volume of traffic (because all CoNTra transmissions intersect around D's neighborhood), and any additional nodes picked up as potential receivers will have significantly less traffic. For example, if the path S→3→5→7→8 is used, the suspects are nodes 1, 2, 3, 5, 7, 8, and D[1]. It may be tempting to add S→2→4→6 and split the traffic evenly between the two paths, but this only serves to make the possible traffic at nodes 2 and D (the intersection of the overhearing area of the two paths) double that at the other nodes. Keeping the same path for a given topology ensures that all nodes get the same amount of traffic (and therefore the same amount of suspicion) even if the list of nodes is smaller.

Reliability Packet delivery may be less reliable with CoNTra because any guaranteed delivery or reliability mechanism present in lower layers of the network will only apply for delivering the packet to the *addressee*, not our intended destination. For example, the 802.11 RTS/CTS will not help avoid collisions at D because it is not part of the exchange. Retransmission triggered by acknowledgments would improve delivery probability but would create a data-`ACK` pattern revealing S and D's relationship. To address this while providing guaranteed delivery we use batched negative acknowledgments. This removes both temporal correlation and 1:1 ratio of data and `ACK`s.

Internal Observers A shortcoming of the basic CoNTra technique is some nodes in the network receive extra packets that are useless to them. If an attacker is participating in the network and not just listening passively (an "internal observer") it could receive such unintelligible data, which it may assume is from a CoNTra sender. This would not directly expose the receiver but it would reveal the path being used for CoNTra, narrowing down the set of possible receivers. In the case that S knows of nodes that are *not* attackers S can avoid this problem by addressing CoNTra data to them whenever possible.

If there is no real node in the network that S can safely address packets to it can work with D to create one: S requests a route to some non-existant "ghost node" G, and D responds with a `Route Reply` suggesting it has a direct link to G. S can now address a CoNTra packet to G; D will get the packet and forwards it to G but no real node will receive unexpected messages. The limitation of this method is it can be suspicious for D to be the only node that ever hears G. Also, it is impossible to create a node in the network anywhere else but adjacent to D, which may result in a shorter path and therefore fewer suspect nodes.

Route failures can also provide opportunities for avoiding internal observers. DSR and other protocols use `Route Error` messages to notify other nodes in the network of failed links. If S hears one of these messages concerning a path from which D can overhear, S can send on this path with a very low likelihood of the addressee actually receiving the packet. D will still receive the packet as it is on the near side of the break. Since this technique would be suspicious if S receives a `Route Error` message in response to an attempt to send on a route, S should only do this if it finds out about the error by overhearing a `Route Error` message destined for another node.

III. Conclusions and Future Work

We have outlined a new approach to countering traffic analysis by a pair of nodes, a scenario that is largely unaddressed by previous research on anonymity. Our proposed approach distributes the perceived amount of traffic from a source to a destination among nodes other than the intended destination. We have discussed the benefits and challenges of this approach, though a formal protocol specification remains as our future work. Additionally, we plan to evaluate the efficacy of CoNTra w.r.t. the number of nodes, number of observers, mobility patterns, and traffic volumes. We also intend to implement CoNTra in Click on our testbed of laptops and Meraki wireless routers.

References

[1] CHAUM, D. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM 4*, 2 (February 1981).

[2] GOLDSCHLAG, D. M., REED, M. G., AND SYVERSON, P. F. Hiding routing information. In *Proceedings of the First International Workshop on Information Hiding* (London, UK, 1996), Springer-Verlag, pp. 137–150.

[3] REITER, M., AND RUBIN, A. Crowds: Anonymity for web transactions. *ACM Transactions on Information and System Security 1*, 1 (June 1998).

[4] ZHANG, Y., LIU, W., LOU, W., AND FANG, Y. Mask: Anonymous on-demand routing in mobile ad hoc networks. In *Transactions on Wireless Communications* (September 2006), vol. 21, IEEE, pp. 2376–2385.

[1] Node 8 was chosen as the endpoint because although 7 works as well, using 8 adds an additional node to the set of suspects.